**BENV 7500 – Programmable Cities**

**Assignment 1 (30% of overall course mark)**

**Due 1st May 2018, 11:59 pm.**

Rubric

|  |  |  |  |
| --- | --- | --- | --- |
| Learning outcomes | Assessment items | Exercises | Weight |
| Demonstrate an understanding of programming concepts, methods and approaches. | Understand code sample. | Explain what a given piece of code does. | 10% |
| Able to fix existing code with errors. | Indicate the types of errors in given code and fix the errors. | 20% |
| Design and implement simple solutions in Python for data analysis tasks. | Able to develop a new or adapt an existing program to answer given questions. | Based on data provided, answer given questions by re-using and adapting code learned in class. | 30% |
| Apply programming knowledge and skills appropriate to analyse and manipulate data for practical urban issues. | Able to develop data analytics through programming applied to relevant urban issues. | Based on data provided, answer given questions by developing new code and plots. | 40% |

Assignment 1 is provided to you as a Jupyter Notebook and should be completed in that format.

The Assignment consists of 4 exercises which together cover the course learning outcomes. By developing those exercises, you will demonstrate the knowledge and skills acquired during the first block of classes.

**Exercise 1 (10%):**

In this exercise you will be given a piece of code in Python. You are asked to carefully read the code and then explain what this code does by adding comments to the code. This exercise tests your understanding of the structure and syntax of the Python programming language.

**Exercise 2 (20%):**

In this exercise you will be given three pieces of code in Python with some errors, and a set of values for variables. You are asked to (a) identify the errors; (b) add comments to the code explaining the types of errors found; (c) fix the errors; (d) run the corrected code with the given values for variables; and (e) save the notebook with the code run, so the output is included in the notebook. This exercise tests your understanding of the structure and syntax of the Python programming language and your capacity to manipulate it.

**Exercise 3 (30%):**

In this exercise you will be given a dataset and three questions (10% for each question). You are asked to answer those questions through data analysis in Python. You can re-use and adapt code in Python learned in class to develop your analysis and obtain responses to the questions. You must include the code in the notebook and use comments in the code to describe the sequence of procedures and the results obtained for each question. This exercise tests your understanding of Python structure and syntax and your ability to re-use and adapt code for different purposes.

**Exercise 4 (40%):**

In this exercise you will be asked to (a) retrieve a dataset from the CityData web repository, (b) write code to answer 2 questions, and (c) write code to develop plots to support your answers. Your code will be based on the learning in class; it will involve not only adaptation of existing code, but also your first endeavour in developing your own code. Comment your code to describe your sequence of procedures, results, plots, and insights. This exercise tests your ability to use Python programming to explore and visualise data and gain insights to answer questions.

**Exercise 1:**

*Code that (a) reads a file (Census table of population by SA2 for the whole of NSW), then (b) selects records which satisfy a condition (SA2s in Greater Sydney), and then (c) counts the number of SA2s in greater Sydney, (d) sums the total population in those SA2s, (e) identifies the 10 most populated SA2s, and (f) calculates the proportion of the population of Greater Sydney which lives in the 10 most populated SA2s.*

*We need to write code in Python with clear file and variable names, but without comments (students should add comments).*

**Exercise 2:**

Exercise 2(a):

*Code with syntax error (capitalisation/spelling error of name of variable)*

Exercise 2(b):

*Code with syntax error (= and = =)*

Exercise 2(c):

*Code with logic error (equation and brackets) … For example, the conversion of Celsius to Fahrenheit follows the function F = C \* 1.8 + 32 … so 40 degrees Celsius equal 104 Fahrenheit, and 10 degrees Celsis equals 50 F. Write the code with the following wrong equation: F = C \* (1.8 + 32) …*

**Exercise 3: RIDERLOG EXs**

*Opal data is provided containing records of location of tap on and tap off in NSW by time (15 min interval) mode of public transport, weekday, etc. Write code to perform the tasks below, adding comments along your code, indicating tasks being performed. Run the code and save the notebook with the outputs shown. Add markdown cells to explain the outputs.*

*Load this file in your Jupyter notebook and answer the following questions:*

*3a. How many records (rows) and attributes (columns) does this dataset have?*

*3b. Visualise the first few records of this dataset, to understand the attributes available.*

*3c. Print the list of unique modes of transport in the dataset.*

*3d. How are trips distributed by day of the week?*

*3e. Plot the results of question 3d as a bar chart.*

*3f. How many trips are made weekly in NSW by each mode of transport?*

*3g. Make a pie chart of your result for question 3f.*

*What else using conditional and loop?????*

*Could do a for loop through the data…hmmm but why….*

import numpy as np

import pandas as pd

df1 = pd.read\_csv("F:/1 Programmable Cities/Assignment 1/OpalCounts\_AllModes\_Region\_Aug16 .csv", low\_memory=False) #Load file

df1.shape #Check nb rows and columns (3a) … answer: (217248, 11)

df1.head() #See top records (3b)

print (df1.Mode.unique()) #print list of unique values for field 'Mode' (3c) … answer: ferry, train, lightrail and bus

df1\_byday = df1.groupby('Weekday')['Count'].sum( ) #sum trips by day of the week (3d)

print(df1\_byday) #answer: Mon 3.98M, Tue 40.7M, Wed 40.7M, Thu 41.1M, Fri 39.9M, Sat 1.85M, Sun 1.50M)

[[[[Jonathan code for bar chart]]]]] import plot lib …. 3e

df1\_weeklybymode = df1.groupby('Mode')['Count'].sum() #sum of weekly trips by mode of transport (3f)

print(df1\_weeklybymode) #answer: Bus 9.75M, ferry 194k, lightrail 194k, train 13.45M

[[[[Jonathan code for pie chart]]]]] import plot lib …. 3g

????

**Exercise 4:**

*Harvest Opal data from CityData*

*Questions???*

*Opal data is provided containing records of location of tap on and tap off in NSW by time (15 min interval) mode of public transport, weekday, etc. Write a code to perform the tasks below, adding comments along your code, indicating tasks being performed and results obtained.*

*Load this file in your Jupyter notebook and answer the following questions:*

*3a. How many records (rows) and attributes (columns) does this dataset have?*

*3b. Visualise the first few records of this dataset, to understand the attributes available.*

*3c. Print the list of unique modes of transport in the dataset.*

*3d. How are trips distributed by day of the week?*

*3e. Plot the results of question 3d as a bar chart.*

*3f. How many trips are made weekly in NSW by each mode of transport?*

*3g. Make a pie chart of your result for question 3f.*

*What else using conditional and loop?????*